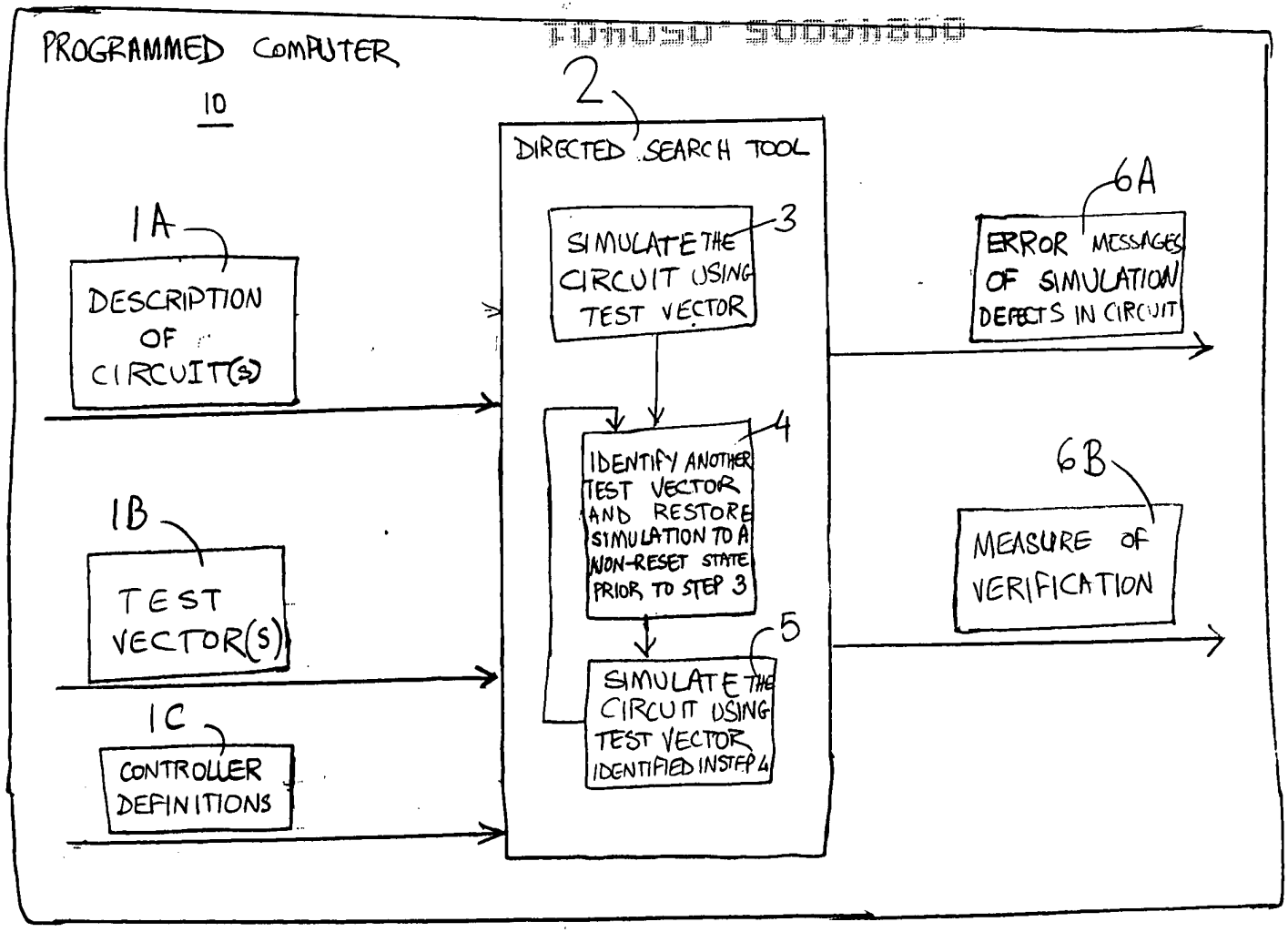
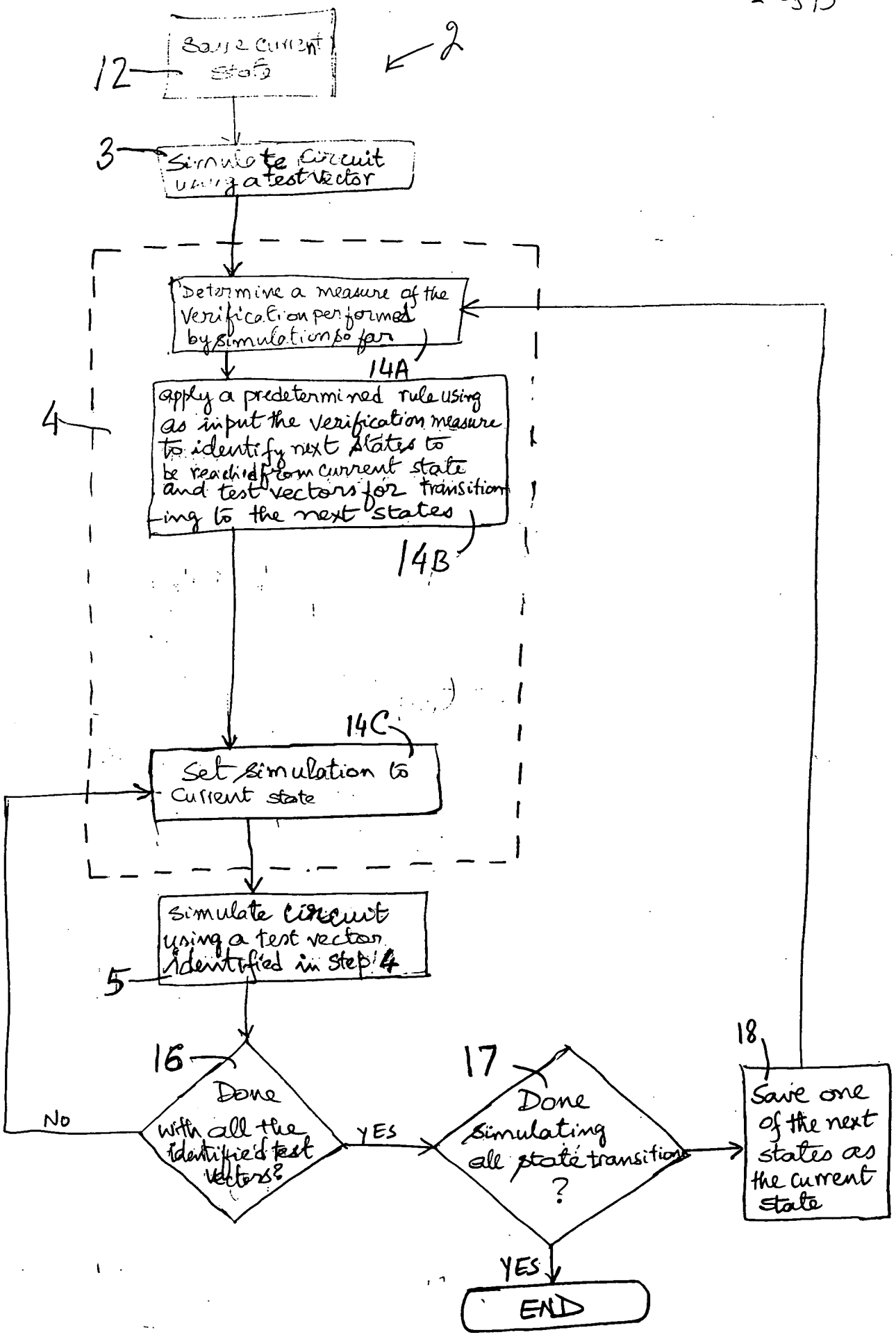

PROGRAMMED COMPUTER
10
2
DIRECTED SEARCH TOOL
1A
DESCRIPTION OF CIRCUIT(S)
1B
TEST VECTOR(S)
1C
CONTROLLER DEFINITIONS
3
SIMULATE THE CIRCUIT USING TEST VECTOR
4
IDENTIFY ANOTHER TEST VECTOR AND RESTORE SIMULATION TO A NON-RESET STATE PRIOR TO STEP 3
5
SIMULATE THE CIRCUIT USING TEST VECTOR IDENTIFIED IN STEP 4
6A
ERROR MESSAGES OF SIMULATION DEFECTS IN CIRCUIT
6B
MEASURE OF VERIFICATION

FIG. 1A


Save current state
12
2
3
Simulate circuit using a test vector
Determine a measure of the verification performed by simulation so far
14A
4
apply a predetermined rule using as input the verification measure to identify next states to be reached from current state and test vectors for transitioning to the next states
14B
14C
Set simulation to current state
simulate circuit using a test vector identified in step 4
5
16
Done with all the identified test vectors?
No
YES
17
Done simulating all state transitions ?
18
Save one of the next states as the current state
YES
END

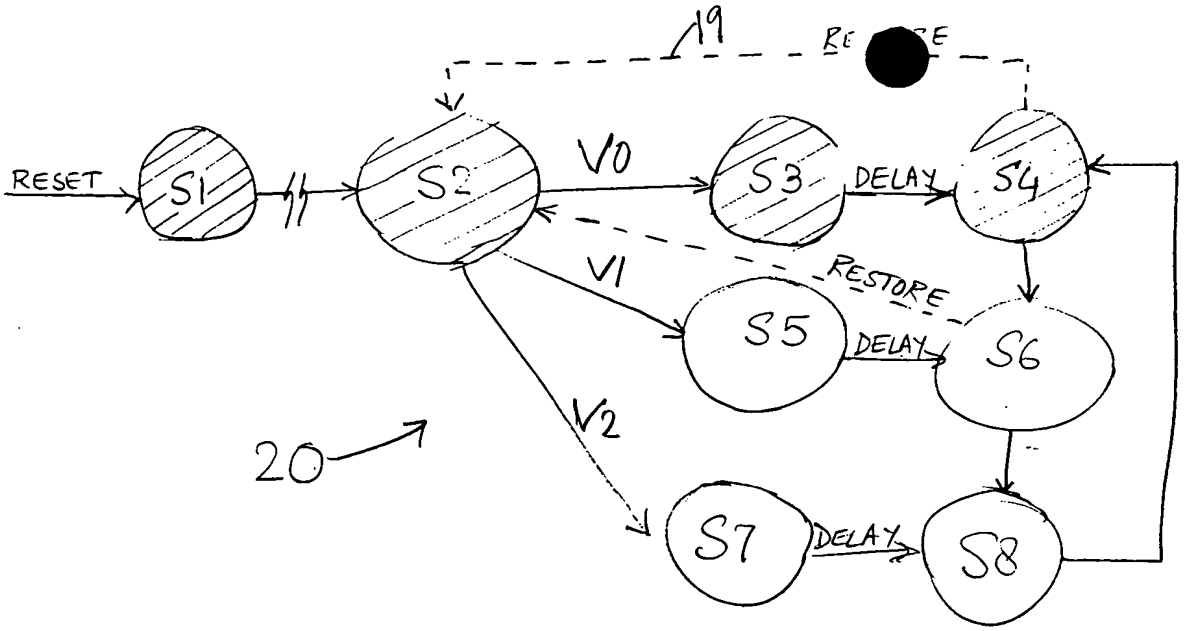

19
RESET
S1
S2
V0
S3
DELAY
S4
RESTORE
V1
S5
DELAY
S6
V2
20
S7
DELAY
S8

FIG. 1C

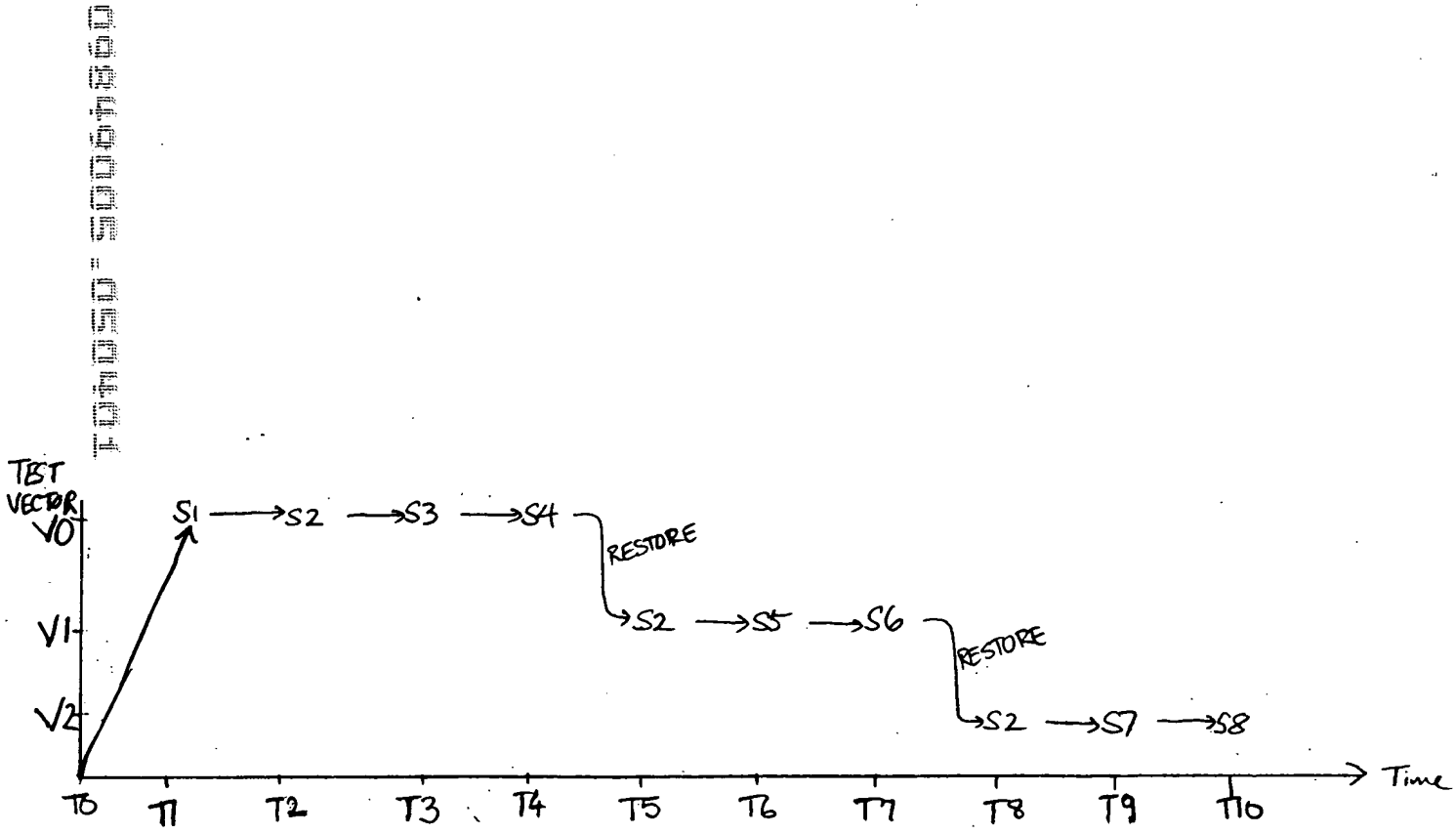

TBT VECTOR
V0
V1
V2
S1 → S2 → S3 → S4
RESTORE
S2 → S5 → S6
RESTORE
S2 → S7 → S8
Time
T0
T1
T2
T3
T4
T5
T6
T7
T8
T9
T10

FIG. 1D

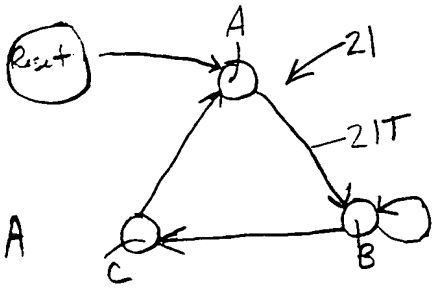


FIG 2A

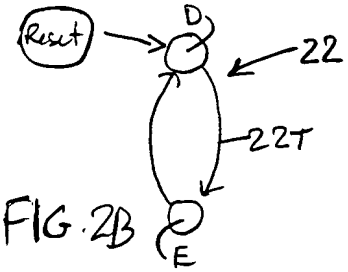


FIG. 2B

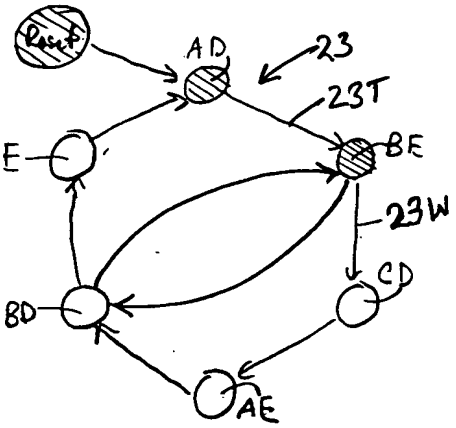


FIG. 2C

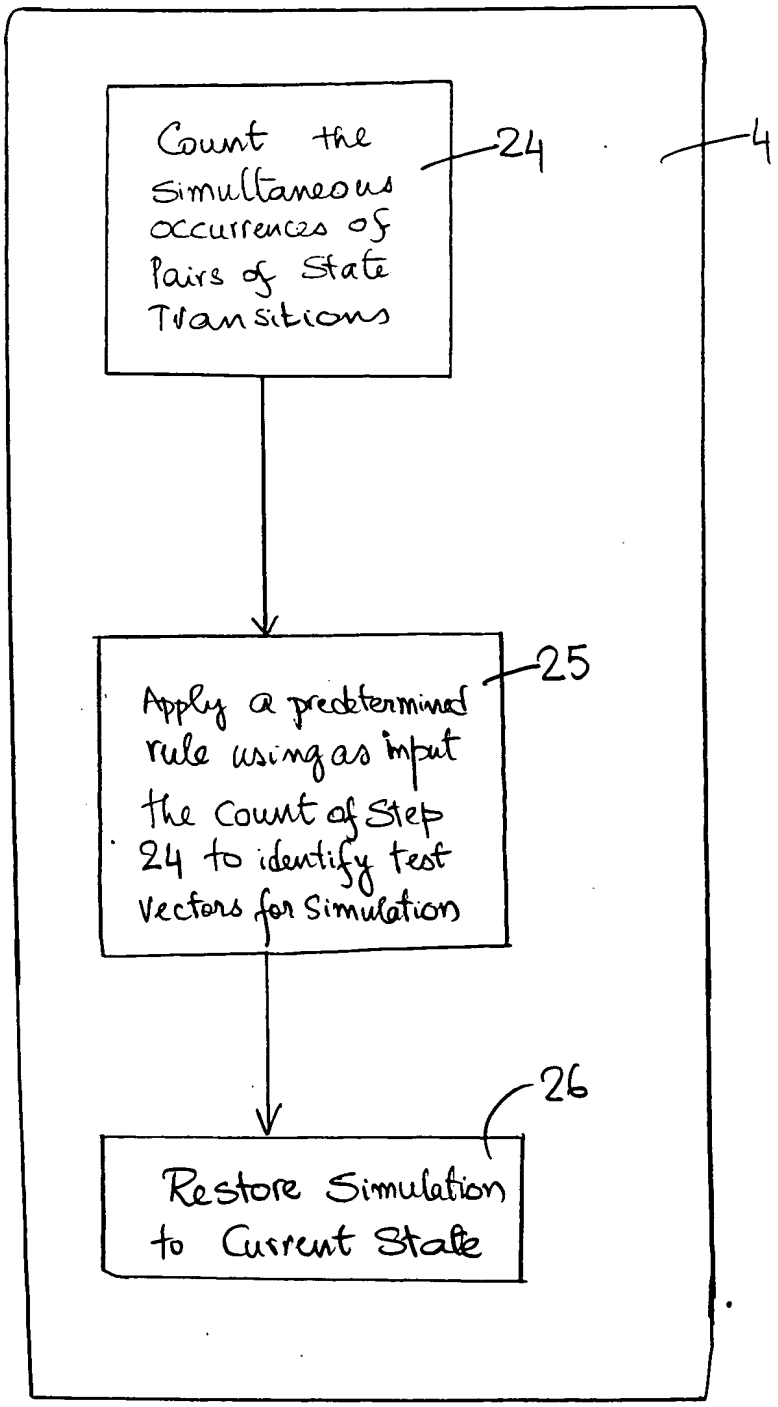

Count the simultaneous occurrences of pairs of state transitions
24
4
Apply a predetermined rule using as input the count of step 24 to identify test vectors for simulation
25
Restore simulation to current state
26

FIG. 2D

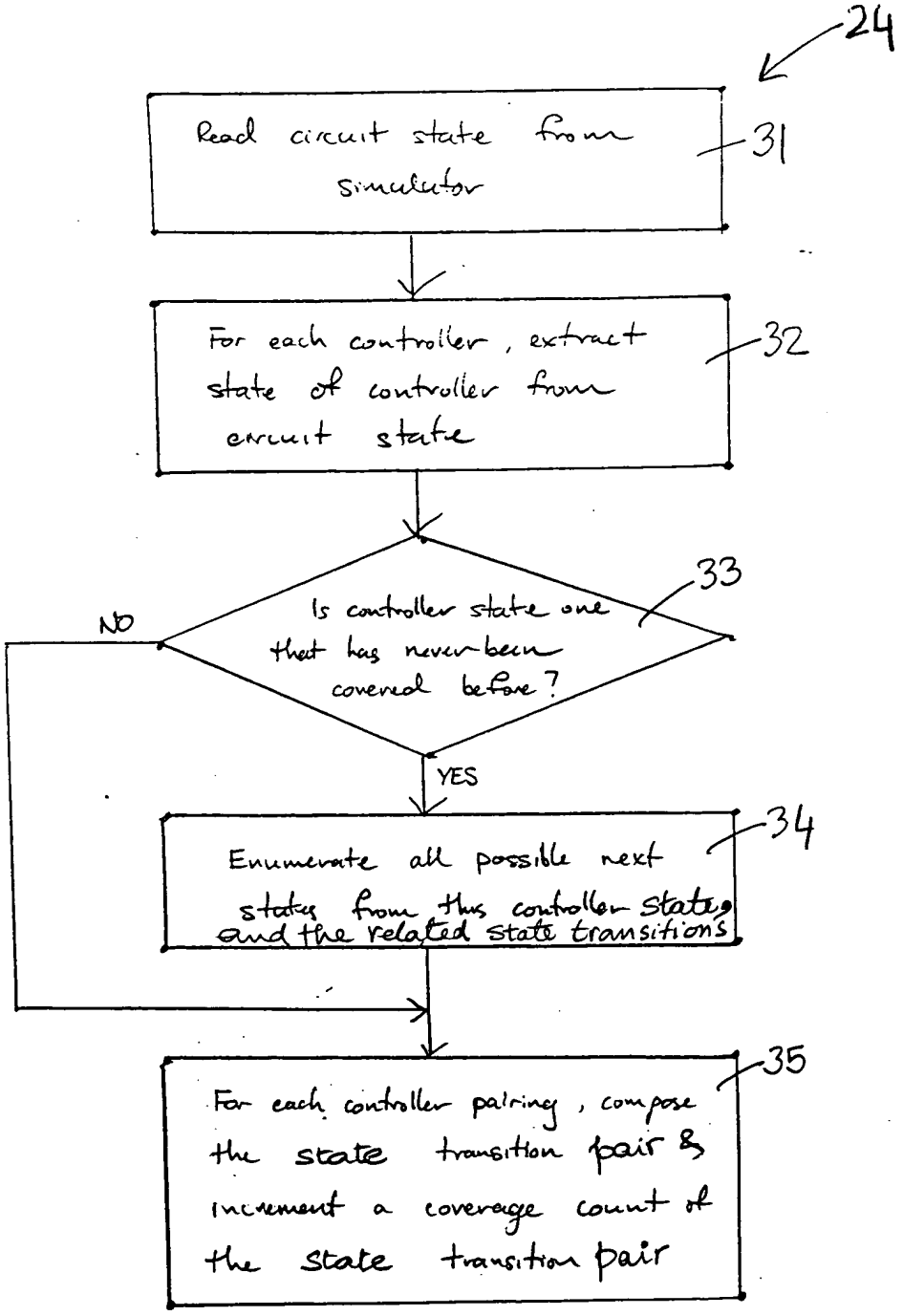

24
Read circuit state from simulator
31
For each controller, extract state of controller from circuit state
32
Is controller state one that has never been covered before?
33
NO
YES
Enumerate all possible next states from this controller state, and the related state transitions
34
For each controller pairing, compose the state transition pair & increment a coverage count of the state transition pair
35

FIG. 2E

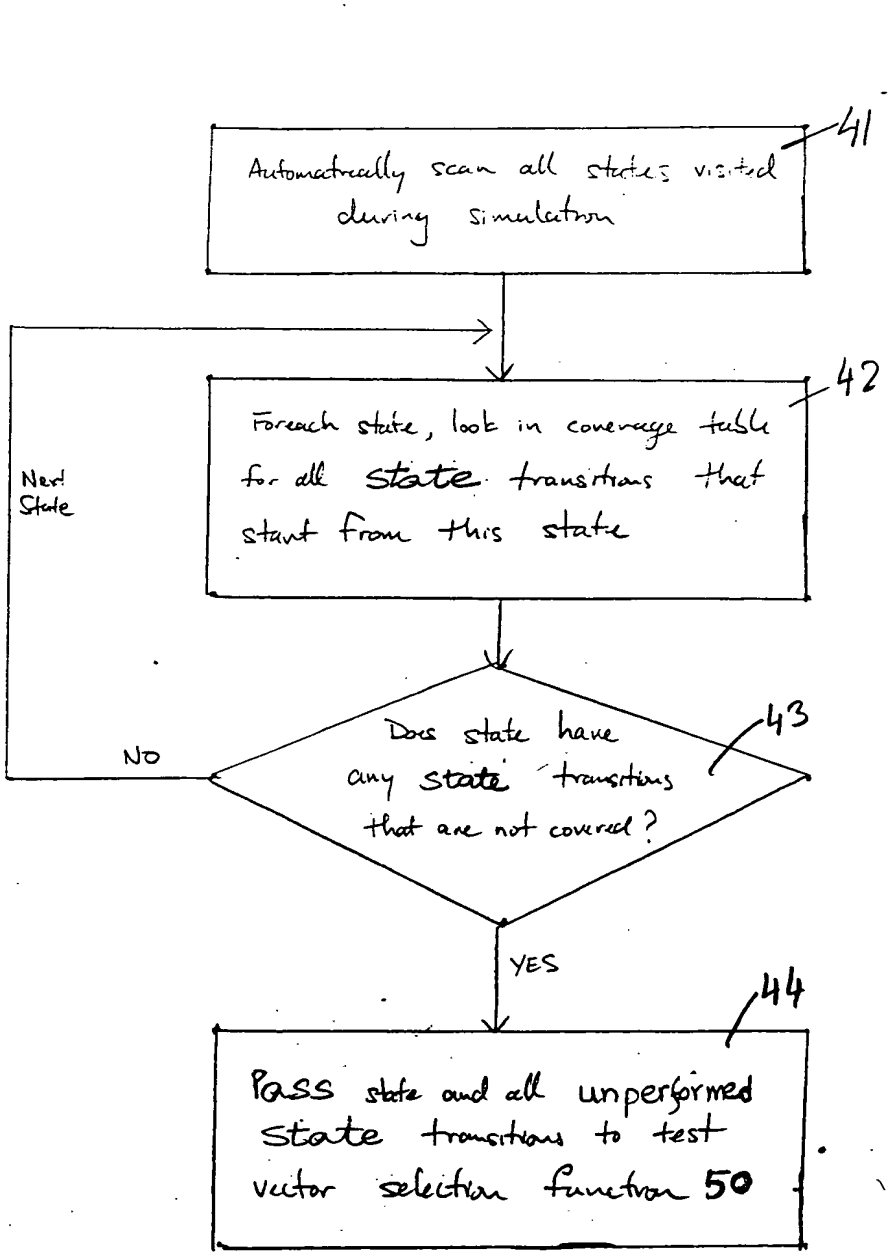

25
41
Automatically scan all states visited during simulation
42
For each state, look in coverage table for all state transitions that start from this state
Next State
43
Does state have any state transitions that are not covered?
NO
YES
44
Pass state and all unperformed state transitions to test vector selection function 50

FIG. 2F

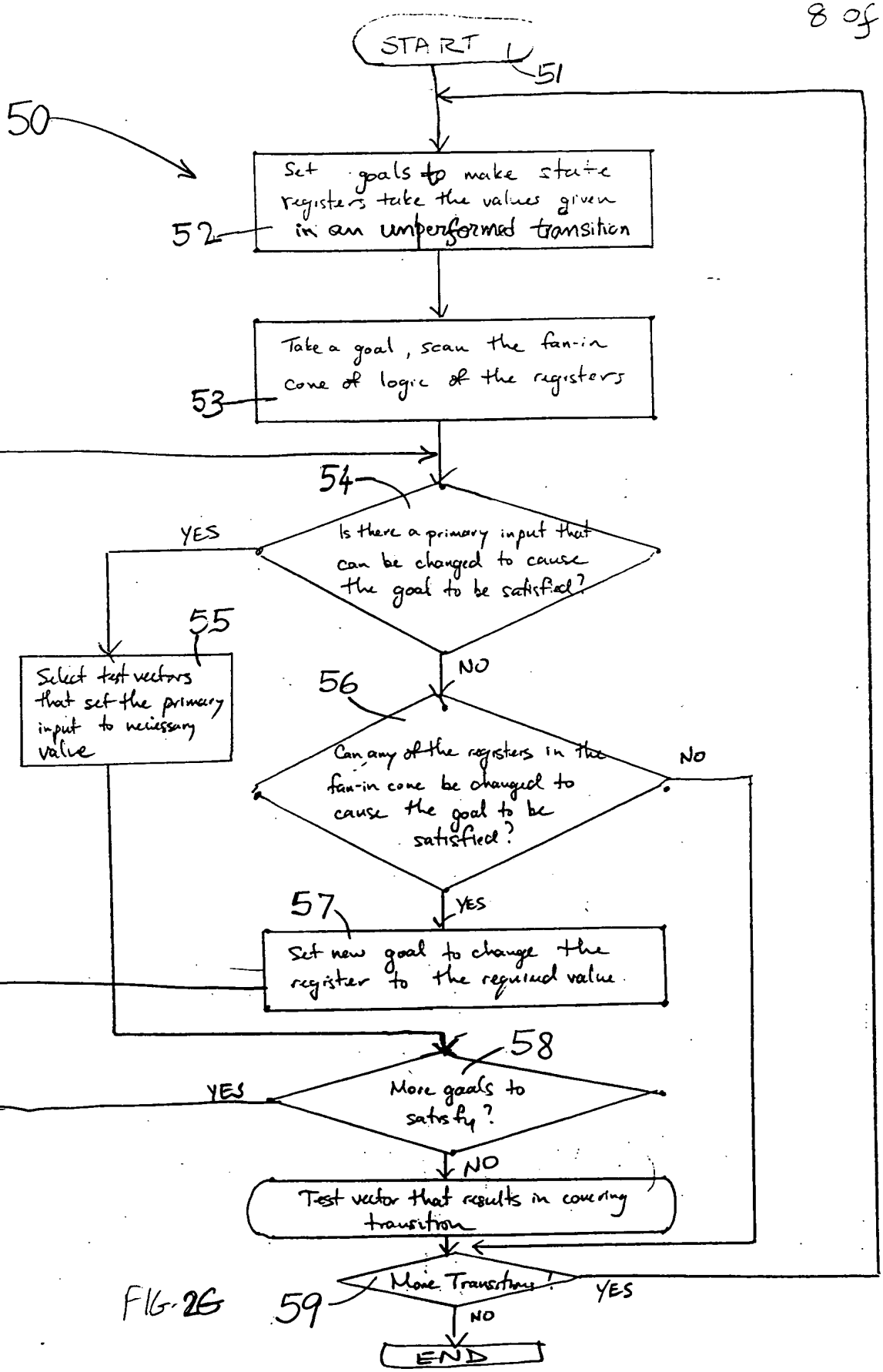
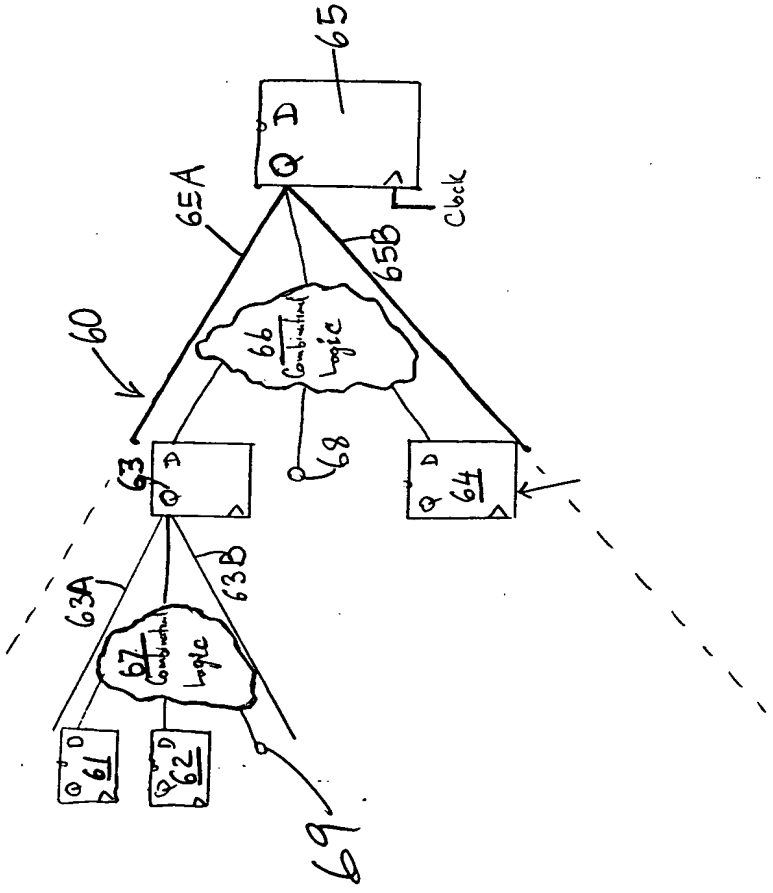

START
51
50
Set goals to make state registers take the values given in an unperformed transition
52
Take a goal, scan the fan-in cone of logic of the registers
53
54
Is there a primary input that can be changed to cause the goal to be satisfied?
YES
NO
55
Select test vectors that set the primary input to necessary value
56
Can any of the registers in the fan-in cone be changed to cause the goal to be satisfied?
NO
YES
57
Set new goal to change the register to the required value
58
More goals to satisfy?
YES
NO
Test vector that results in covering transition
More Transitions?
59
YES
NO
END

FIG. 26

FOTOSO: 20070290

FIG. 24



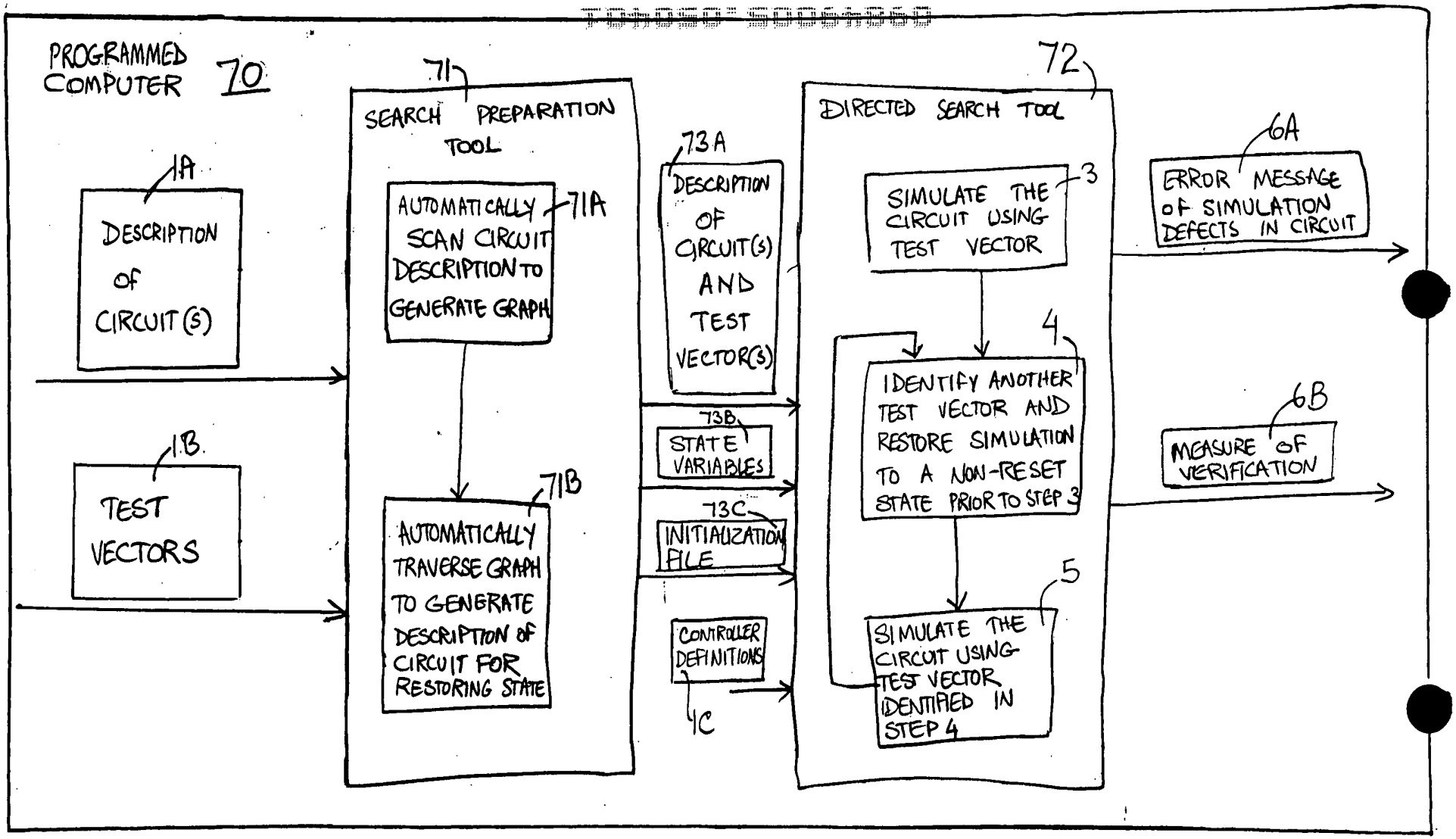

PROGRAMMED COMPUTER 70
71
SEARCH PREPARATION TOOL
72
DIRECTED SEARCH TOOL
1A
DESCRIPTION OF CIRCUIT(S)
1B
TEST VECTORS
AUTOMATICALLY SCAN CIRCUIT DESCRIPTION TO GENERATE GRAPH
71A
AUTOMATICALLY TRAVERSE GRAPH TO GENERATE DESCRIPTION OF CIRCUIT FOR RESTORING STATE
71B
73A
DESCRIPTION OF CIRCUIT(S) AND TEST VECTOR(S)
73B
STATE VARIABLES
73C
INITIALIZATION FILE
CONTROLLER DEFINITIONS
4C
SIMULATE THE CIRCUIT USING TEST VECTOR
3
IDENTIFY ANOTHER TEST VECTOR AND RESTORE SIMULATION TO A NON-RESET STATE PRIOR TO STEP 3
4
SIMULATE THE CIRCUIT USING TEST VECTOR IDENTIFIED IN STEP 4
5
6A
ERROR MESSAGE OF SIMULATION DEFECTS IN CIRCUIT
6B
MEASURE OF VERIFICATION

FIG. 3A

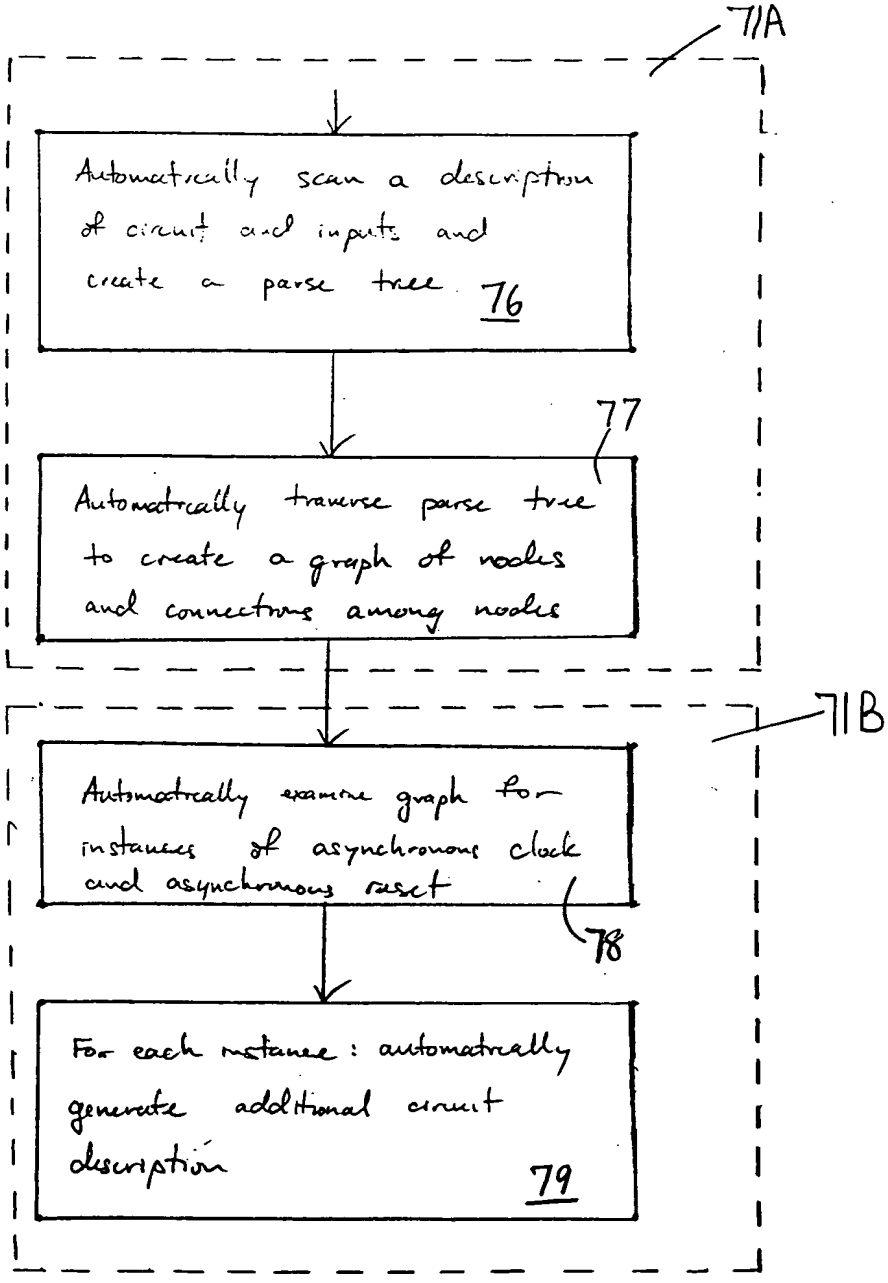

71A
Automatically scan a description of circuit and inputs and create a parse tree 76
77
Automatically traverse parse tree to create a graph of nodes and connections among nodes
71B
Automatically examine graph for instances of asynchronous clock and asynchronous reset
78
For each instance: automatically generate additional circuit description 79

FIG. 3B

FIG. 3C

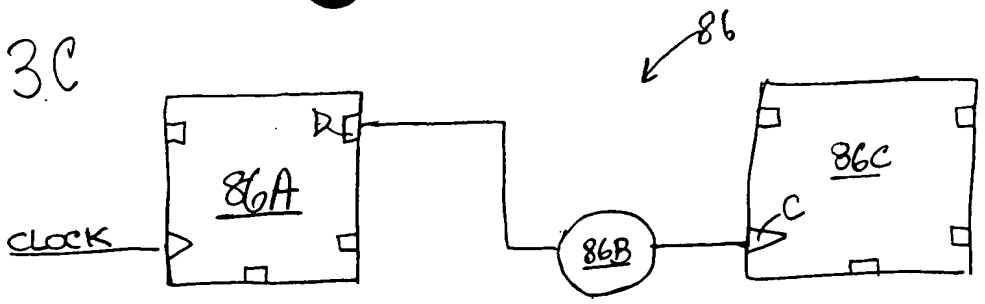


FIG. 3D

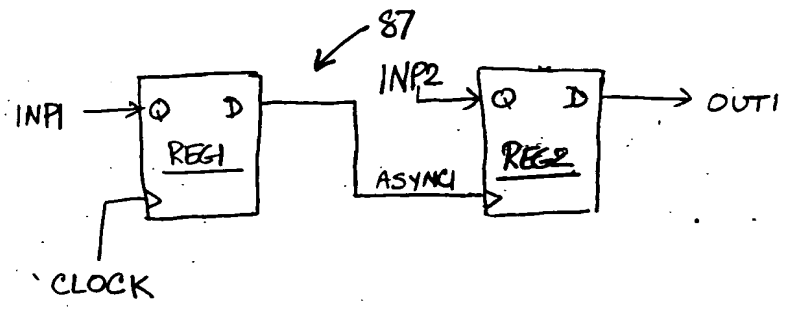


FIG. 3E

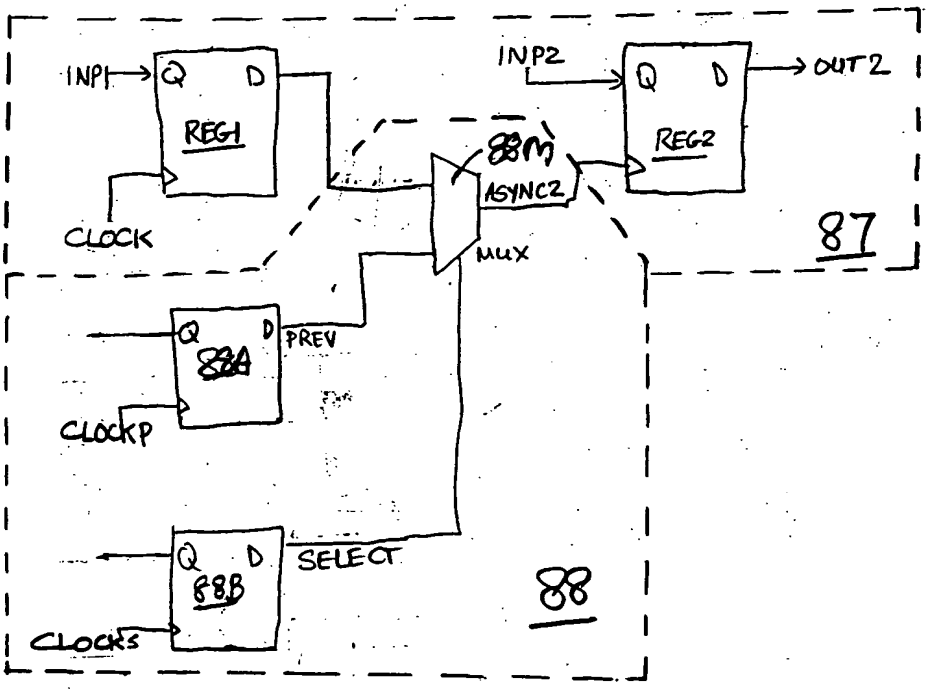
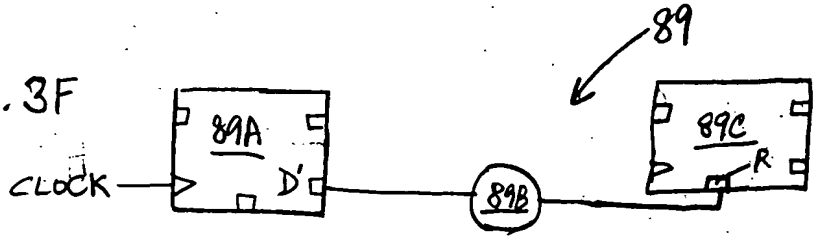


FIG. 3F



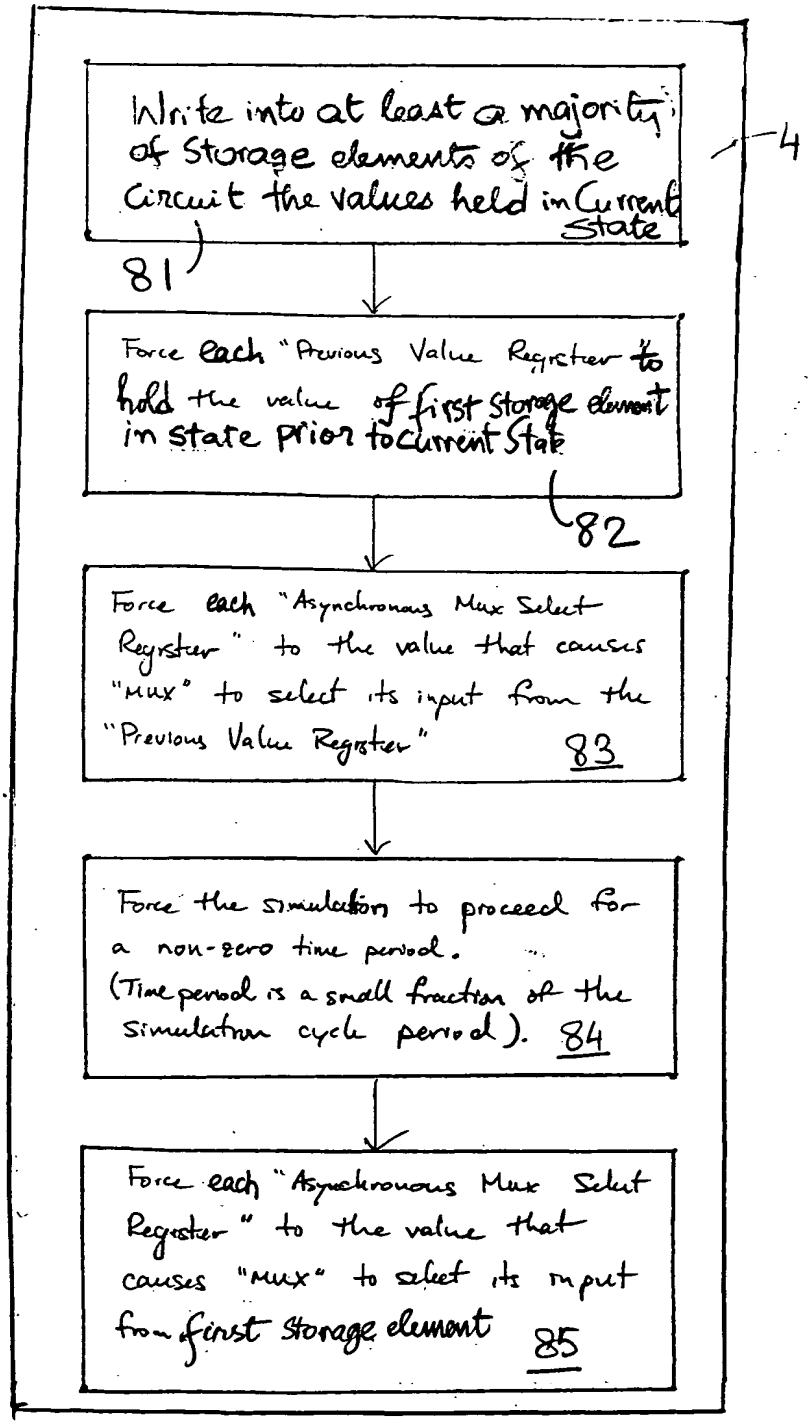

Write into at least a majority of Storage elements of the Circuit the values held in Current State
81
4
Force each "Previous Value Register" to hold the value of first storage element in state prior to current State
82
Force each "Asynchronous Mux Select Register" to the value that causes "Mux" to select its input from the "Previous Value Register"
83
Force the simulation to proceed for a non-zero time period. (Time period is a small fraction of the simulation cycle period).
84
Force each "Asynchronous Mux Select Register" to the value that causes "Mux" to select its input from first storage element
85

FIG. 3G

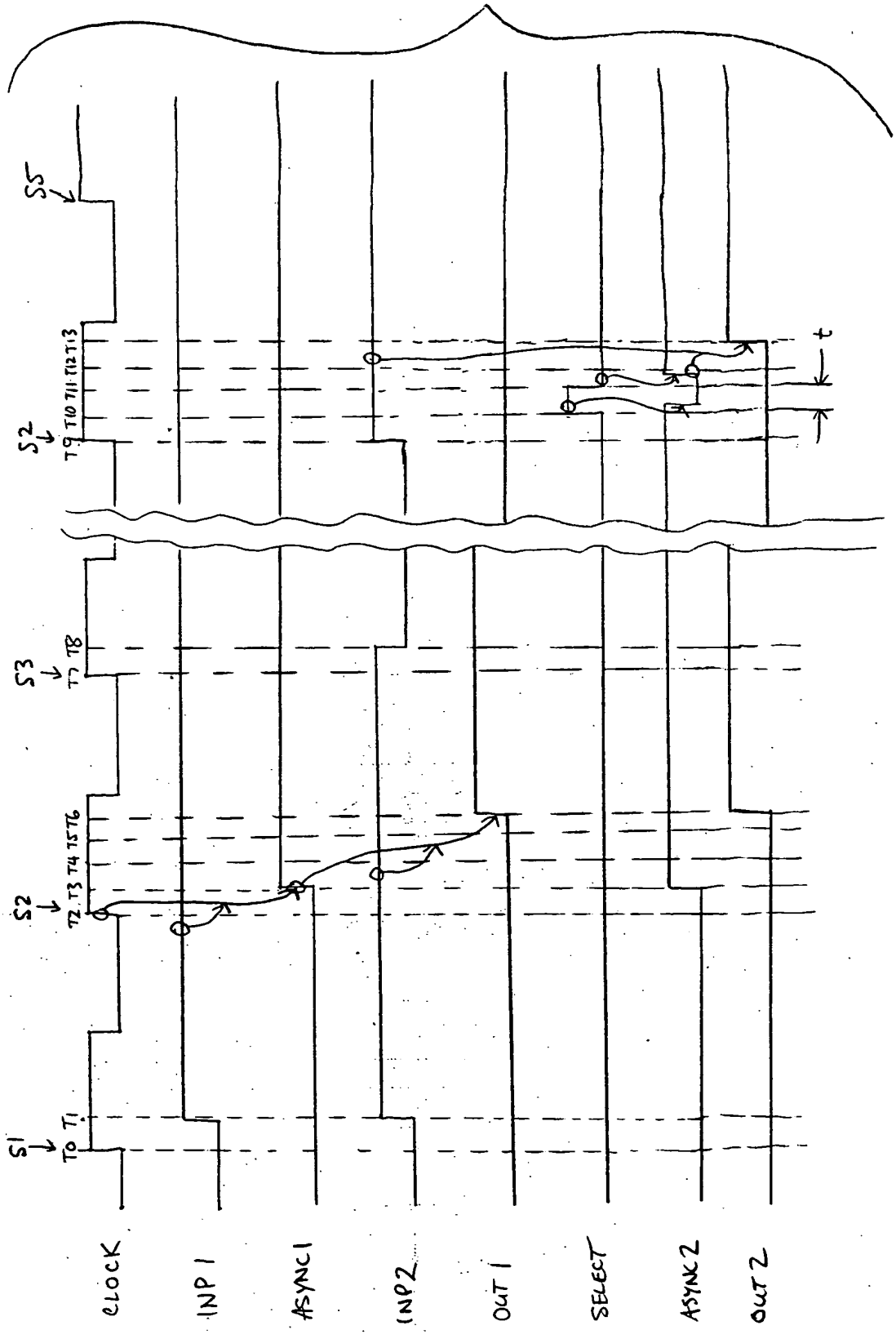

SS
S2
T9 T10 T11 T12 T13
t
S3
T7 T8
S2
T2 T3 T4 T5 T6
S1
To T1
CLOCK
INP 1
ASYNC1
INP2
OUT 1
SELECT
ASYNC2
OUT 2

FIG. 3H

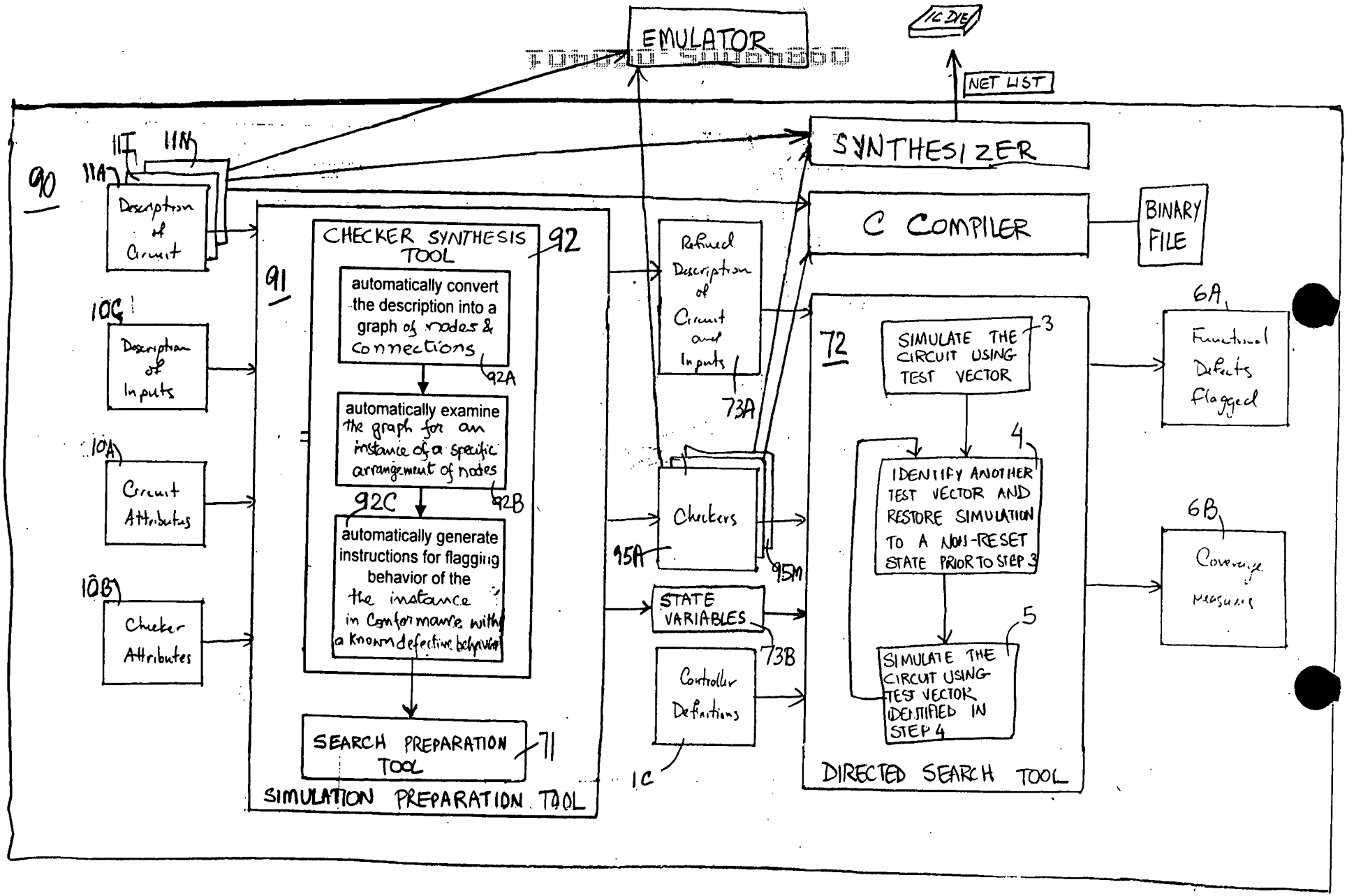


FIG. 4

M-5333
15 of 15